

American International University-Bangladesh (AIUB)  
**Department of Computer Science  
Faculty of Science &Technology (FST)  
Spring 20\_21**

**Section: C  
Group No: 04**

**PROJECT TITLE: TOUR GUIDELINE MANAGEMENT SYSTEM**

A software Engineering project submitted

By

|  |  |  |  |
| --- | --- | --- | --- |
| Serial No. | Student Name | Student ID | Contribution (%) |
| 21 | Sarker, Md. Fazley Rabbi | 19-39444-1 | 25% |
| 25 | Dilruba Khanam Jesey | 19-39979-1 | 25% |
| 26 | Mustary Rahman Prottasha | 19-39985-1 | 25% |
| 38 | Musfiqur Rahman | 19-41056-2 | 25% |

The project will be Evaluated for the following Course Outcomes

|  |  |
| --- | --- |
| Your Project will be Evaluated based on the following marking criteria | Total Marks |
|  |
| Requirements Analysis (functional, quality, and project requirements) [5Marks] |  |
| System Design (UML, UI/UX design) [5Marks] |  |
| Test and Project Management Planning [5Marks] |  |
| Submission, Completeness, Spelling, Grammar and Organization [5Marks] |  |

# PRODUCT AND PROJECT DESCRIPTION

## 1.1 System Features

**Functional Requirements**

## 1.1 The software shall allow user to register with their email id & phone number.

## 1.2 Without registration user can only visit in the home page. There are few options & necessary information in home page. After registration user can able to use other features of the software.

## If the username and/or password has been inserted wrong, the random verification code will be generated and sent to the user’s email address by the system to retry login.

## If the number of login attempt exceed its limit (3 times), the system shall block the user account login for one hour.

## The login credentials (username and password) will be verified with database records.

**Priority Level:** Medium

**Precondition:** User have valid user id, number and password

**Cross-references:**

# Choosing the Best Route

2.1 The software will provide the best possible route for journey according to the selected destination of the users.

* 1. If users select their destination the software will provide the best possible route with the assist of an API called Google Map.

**Priority Level:**High

**Precondition:**User need to provide the destination.

**Cross-references:**

# Finding Ideal Hotel

* 1. The software will allow the users to find ideal hotel in their selected destination. Google Map, API will be used to perform this operation.
  2. If users select a destination then they will need to go to the Ideal Hotel option.
  3. If the user selects Ideal Hotel option a window will appear and show the information from database records and location from the Google Map of those hotels which will be available in selected destination.

**Priority Level:**Medium

**Precondition:**User need to provide destination.

**Cross-references:**

# 4. Nearest Transportations

4.1 The software will allow the user to find the nearest Bus stations, Airports, Railways from the current location of the user.

* 1. User have to turn on the GPS and then select Nearest Transportations.

4.3 Then with the help of Google Map, API the locations of nearest Bus Stations, Airports, Railways will be displayed on the screen.

**Priority Level:**High

**Precondition:**User must turn-on the GPS in the using device.

**Cross-references:**

**5. Weather Update**

5.1The software will provide weather update to the user according to the destination and particular time by using an API named Global Weather.

* 1. User need to go in Weather Update then select the destination. Particular time session can be given otherwise this feature will show the up-to date weather information.

**Priority Level:**High

**Precondition:**User must provide the area.

**Cross-references:**

**6. Create Event**

* 1. The software will allow the user to create event and give a name of the event.
  2. User will need to enter Create Event. The user will need to provide a name for the event.
  3. By performing the tasks of aforementioned options event will be created.
  4. Event creator will be able to handle the privacy of that event by selecting Event Privacy. User can make the event closed or public from there.

User will be able to set the time frame for the event from the option Set Event Time.

* 1. User will be able to approve or cancel the join request of that event as there will be a Join Request window available with option Confirm and Cancel. Join Request option will be found in under the Event Name.
  2. There will be an Event Member List where all of the joined members will be found.
  3. All of the information of the event such as event time, event name, members who will join the event will be stored in the database.
  4. User can delete an existing event by going on Event History. Then find the event name. After that, click Delete.

**Priority Level:** High

**Precondition:** Null

**Cross-references:**7.1, 7.2

# Make Your Groups

7.1 The software will allow the user to make groups among the joined event members.

7.2 User can find other from the Event Member List of Create Event. From there user can make groups among them.

7.3 When user will be making a group with others, a notification will be sent to those whom the user wanted to be joined in the group.

* 1. Only if those members accept the request then they will be added in the group. The record will be stored in the database.
  2. User can control the admin panel by go to Settings of the particular feature.
  3. Admin will be able give permission to join or remove anyone from the group.
  4. Members of the group will be able share their thought by using another feature Chatting.

**Priority Level:** High

**Precondition:** Member need to be joined in the particular event.

**Cross-references:**6.7, 6.8, 8.1, 8.2

# Chatting

* 1. The software will allow the userto chat with other members of the group personal or in the group by selecting Chatting.

8.2 From there the user can find others who are already added in the group. From database this feature will provide the members who are in the group.

* 1. User will be able text others, give voice message, send reply of a particular message, share links, send photos, short videos of length not exceeding the size 25MB. For performing those tasks all of the icons of those sub-features will be on the top of the chat bar.
  2. Besides those the user will be able to mute someone, mute groups, leave groups, creates poll, send emoji, give reply to a particular message with an emoji. These tasks will be performed by selecting those particular sub features and those features will be available in that window.

**Priority Level:** High

**Precondition:** Members need to be present in the group.

**Cross-references:**7.4

# Gallery

9.1 The software will allow the user to use gallery facility to store photos, videos and   files.

9.2 There will be an option Your Memories under the Gallery section.

9.3 If the user wants to upload something then, there will be options from where the user wants to upload the files. User will need to select the files from his desired location.

9.4 User will be able make folders in the gallery. User can name those folders. To performs these tasks user can go to Gallery, under the gallery user will find Create Folders. User can provide the folder names from there.

9.5 After selecting files and the folders there will be an option Upload in the bottom of the window. If the user clicks it the files will be uploaded.

9.6 The user will always be able to delete or add new files in the gallery as all the record and information will be stored in the database.

**Priority Level:** High

**Precondition:** Null

**Cross-references:**Null

# Log Out

**10.1**The software will allow the user to exit the system by using Logout feature.

**10.2**The user will find Logout on top of every window, after once logged in.

**10.3**By clicking Logout the user will find another window.

**10.4**That window will come with two options Exit and Cancel with a message.

**10.5**If the user click cancel, the use user will be still logged in.

**10.6**If the user click exit, the user will be able to exit the system.

**Priority Level:** High

**Precondition:** User must be logged in, in the system.

**Cross-references:**This feature is connected with Log in feature.

# Writing Vlogs

* 1. The software will allow the user to write vlogs in particular topics. By entering Writing Vlogs, the user will get a window form where the user can choose a template for writing vlogs.
  2. After choosing the template the user will be able start writing vlogs right way.
  3. The user can make the vlog public form the privacy settings of this particular feature.
  4. If any vlog is posted by the user other users will be able make comments in the comment section of that vlog.
  5. Other users will be able to give rating for that vlog from the option Rate this Vlogs.

**Priority Level:** High

**Precondition:** Null

**Cross-references:**Null

## 1.2 System Quality Attributes

**Non-Functional Requirements**

* **Availability**

Availability is a measure of the planned uptime during which the system is actually available for use and fully operational.

**AV-1.** The system shall be at least 97 percent available on weekdays between 6:00 a.m.

and midnight local time, and at least 99 percent available on weekdays between 10:00 p.m. and 4:00 p.m. local time.

**Priority Level:** High

* **Performance**

Performance requirements define how well rapidly the system must perform specific functions.

* Speed
* Throughput
* Capacity
* Timing

**PE-1.** Every Web page shall download in 15 seconds or less over a 50 KBPS modem

connection.

**Priority Level:** High

* **Efficiency**

Efficiency is a measure of how well the system utilizes processor capacity, disk space, memory, or communication bandwidth.

**EF-1.** At least 65 percent of the processor capacity and RAM available to the application shall be unused at the planned peak load conditions.

**Priority Level:** Medium

* **Flexibility**

Flexibility measures how easy it is to add new capabilities to the product.

**FL-1.** A maintenance programmer who has at least six months of experience supporting this product shall be able to make a new copy output available to the product, including code modifications and testing, with no more than one hour of labor.

**Priority Level:** High

* **Integrity**

Integrity which encompasses security, deals with blocking unauthorized access to system functions, preventing information loss, ensuring that the software is protected from virus infection, and protecting the privacy and safety of data entered into the system. Integrity is a major issue with Internet software.

**IN-1.** Only users who have Auditor access privileges shall be able to view customer transaction histories.

**Priority Level:** High

* **INTEROPERABILITY**

Interoperability indicates how easily the system can exchange data or services with other systems to provide one-stop services to the customers/users.

**IO-1.** The Chemical Tracking System shall be able to import any valid chemical structure from the chemical draw.

**Priority Level:** High

* **Testability**

Testability refers to the ease with which software components or the integrated product can be tested to look for defects.

**TE-1.** The maximum cyclomatic complexity of a module shall not exceed 20.

* **Useability**

Usability measures the effort required to prepare input for, operate, and interpret the output of the product ν Also referred to as ease of use and human engineering, usability addresses many factors that constitute what users often describe as user-friendliness.

A trained user shall be able to submit a complete request for a chemical selected from a vendor catalog in an average of four and a maximum of six minutes.

**US-1.** A trained user shall be able to submit a complete request for a chemical selected from a vendor catalog in an average of four and a maximum of six minutes.

**Priority Level:** Medium

* **Reusability**

Reusability indicates the relative effort involved to convert a software component for use in other applications. Developing reusable software costs considerably more than creating a component that you intend to use in just one application. Developing reusable software costs considerably more than creating a component that you intend to use in just one application.

**RU-1.** The chemical structure input functions shall be designed to be reusable at the object code level in other applications that use the international standard chemical structure representations.

**Priority Level:** Medium

* **Reliability**

The probability of the software executing without failure for a specific period of time is known as reliability.

**RE-1.** No more than ten experimental runs out of 1000 can be lost because of software Failures.

**Priority Level:** Medium

* **PORTABILITY**

The effort required to transfer a program from one hardware and/or software environment to another

**PO-1.** The software should run in Windows-Android & Mac .

**Priority Level:** Medium

* **Robustness**

Robust software recovers gracefully from problem situations and is forgiving of user mistakes.

**RO-1.** If the editor fails before the user saves the file, the editor shall be able to recover all changes made in the file being edited up to one minute prior to the failure the next time the same user starts the program.

**Priority Level:** High

## 1.3 Project Requirements

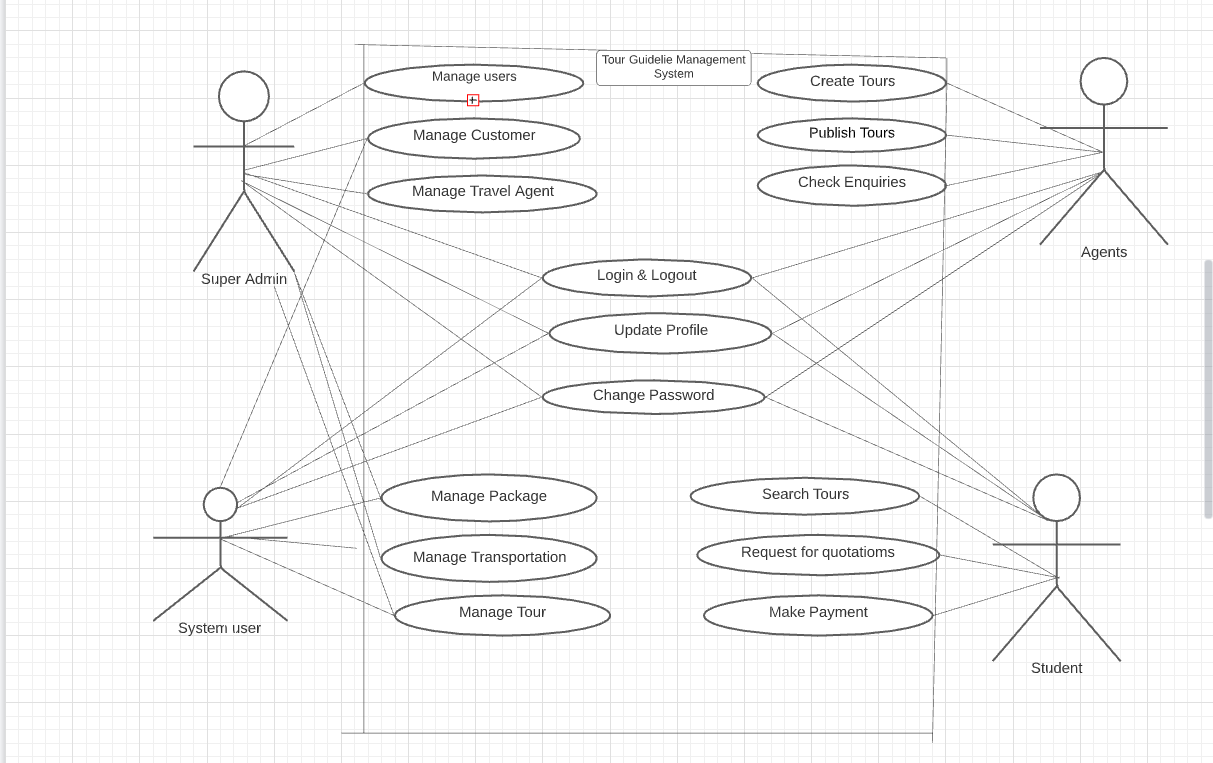
Since, we are developing this project using Scrum so, we will need a team of nine members. As, Scrum is in general Agile practice so, we will need open workplace environment so that, seamless project management can be practiced.

# SYSTEM DESIGN SPECIFICATION

## 2.1 System Design (UML)

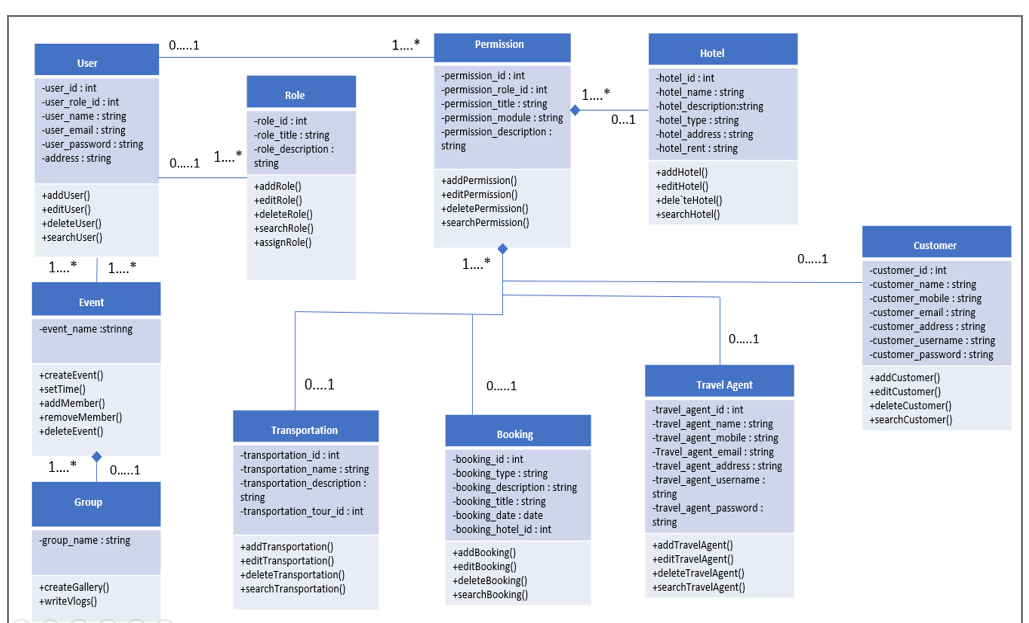
**Use-Case Diagram:**

This use case diagram is a graphic depiction of the interactions among the elements of Tour Guideline System. It represents the methodology used in system analysis to identify, clarify and organize system requirements of Tour Guideline Management System. The main actors of the Tour Guideline Management System in this Use Case Diagram are: Super Admin, System User, Agents, Student who perform the different type of use cases such as Customer, Manager Travel Agent, Manage Package, Manage Transportation, Manage Booking, Manage Hotel, Manage Tour, Manage Users and full Tour Guideline Management System Operations. Major elements of the UML use case diagram of Tour Guideline Management System are shown on the picture below.



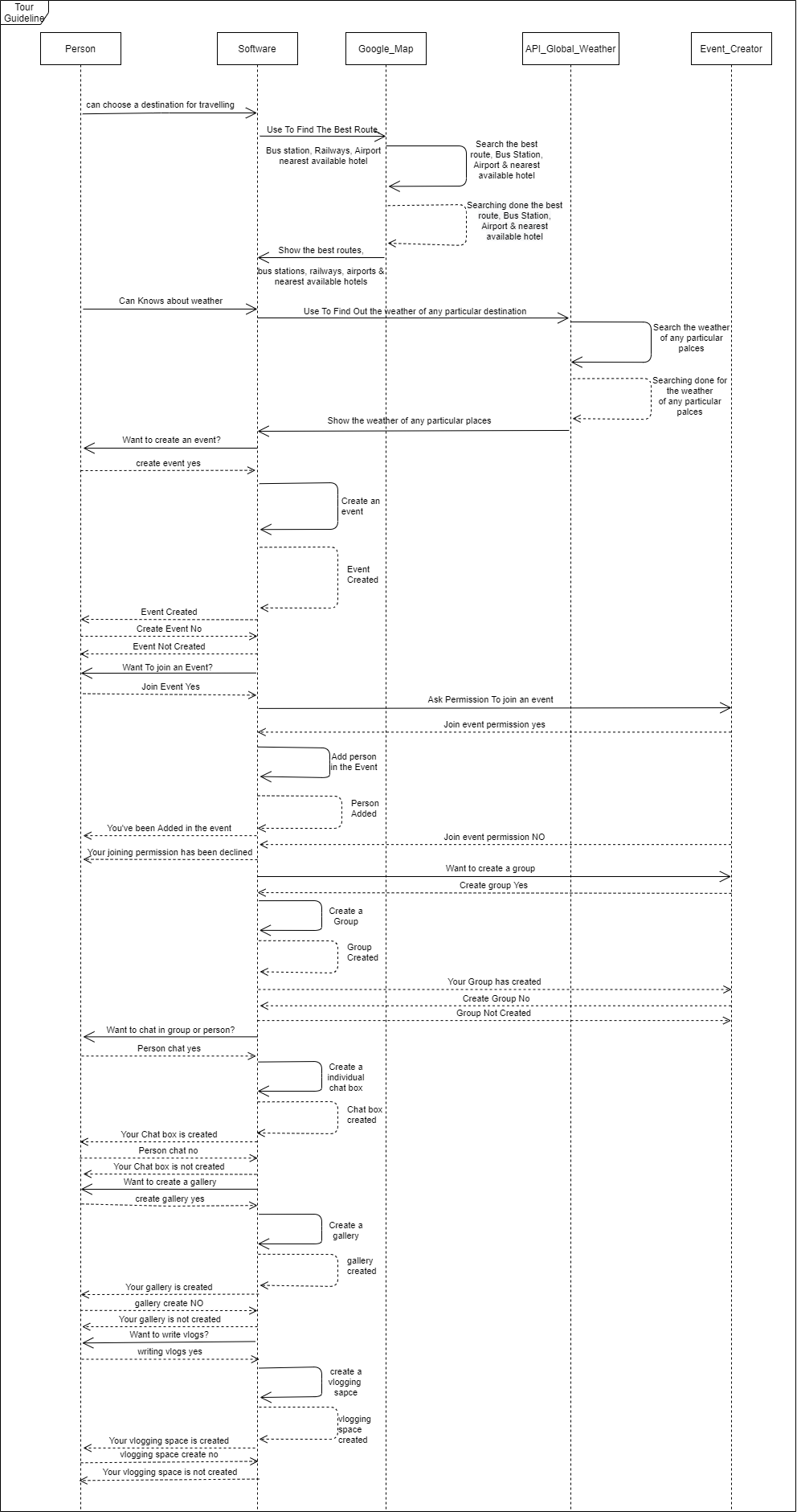
**Class Diagram :**

In our Tour Guideline System there are 10 classes. User can participate more than one event & in an event there may have many user. In an event there may have some group but one group belongs to only one event. User may have role but in one role there may be some user. User need permission & can take only one permission but a permission class can give permission to many user. Same as user class other classes like Transportation class, Booking class, Travel agent class, Customer class & Hotel class can take permission ones from Permission Class but Permission class can give permission many times.



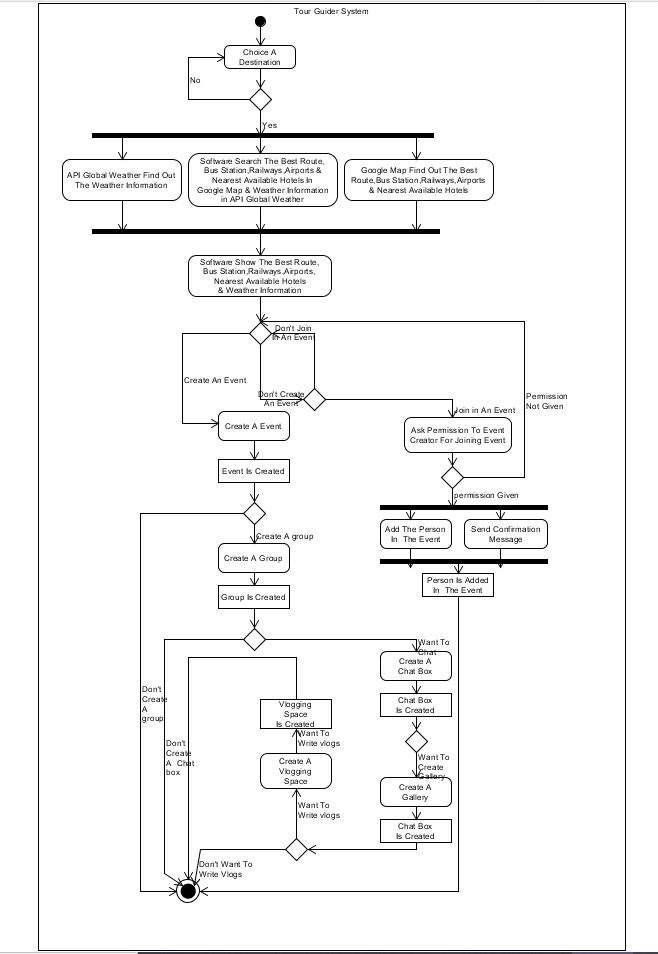
**Sequence Diagram:**

Tour guides & touristswill request to log in into the system for verifying his/ her identity. System will confirm if the account exists or not. If the login verification is unsuccessful, the system notifies that no account found and request to try again. If the login verification is successful then they will be able to use other features. For create or update profile user have to provide necessary documents & information such as NID/ Birth Certificate, Profile Photo, Phone Number, Email, Home address. If those information are valid then it will be stored in database else not. User can search for information and if it is available in database then database will provide search information. For other features, if a tourists wants to know about the weather condition of one tourists spot, system will auto search for the weather in google and show it to the user. In the same way if a tourist or a user looks for the way or try to find the place or bus station, car parking etc in GPS, system will show the best routes & location from user’s current location by using google map. Also user can make a gallery by uploading picture. User can create an event by himself and he/she will be the creator of the event. If any other users want to join that event, they have to ask permission for joining that event from the even creator. If event creator approve their request they can join the event. Event creator will have the access to cancel the event.



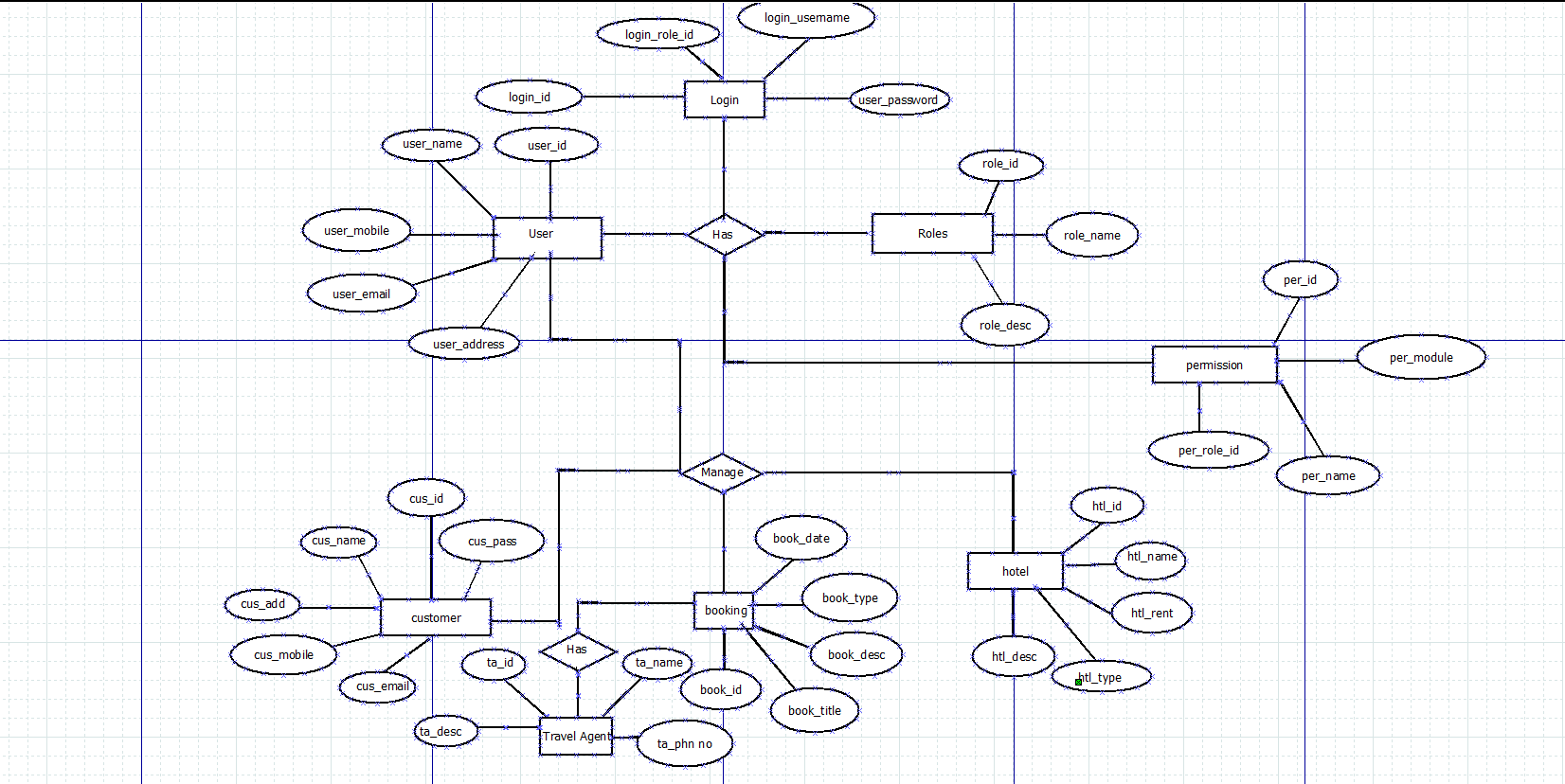
**Activity Diagram:**

At first user have to select a destination. After finding the destination API global weather will find out the weather condition of the destination place, software will provide best routes, bus station airports, railways, nearest hotels & weather condition from google maps and API Global Weather. After that user can create an event or join an on going event. If user wants to create an event he/she will be the event creator and he/she hast to make a group where user & group members can vlog. If the user don’t want to create an event then they have to join an existing event. He/she have to apply for joining an event, if event creator approve the request then the he/she will get a confirmation message & the requested user can join the group and vlog. In the group they can create chat box and a gallery where group members can chat or upload photo in gallery.



**E-R Diagram:**

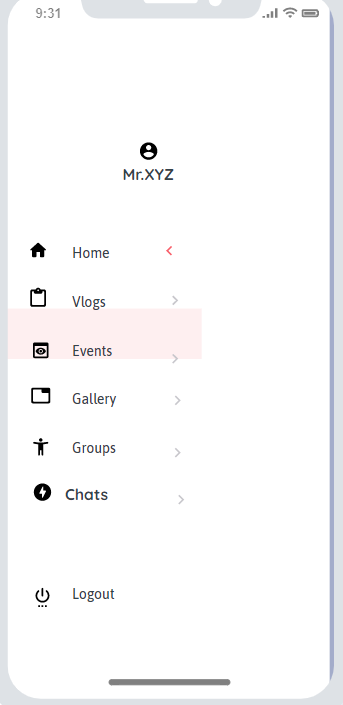
This E-R (Entity Relationship) Diagram represents the model of Tour Guideline System Entity. The entity-relationship diagram of Tour Guideline System shows all the visual instrument of database tables and the relations between Travel Agent, Transportation, Customers, Hotel etc. It used structure data and define the relationships between structured data groups of Tour Guideline System functionalities. The main entities of the Tour Guideline System are Customer, Travel Agent, Package, Transportation, Booking and Hotel.

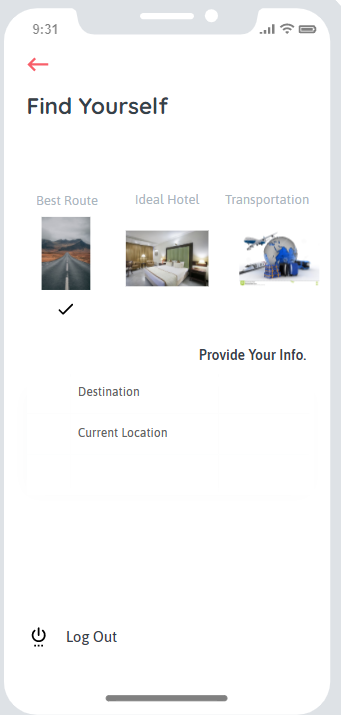


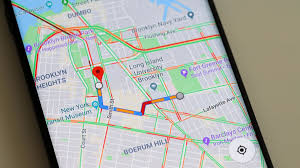
## 2.2 UI/UX Design

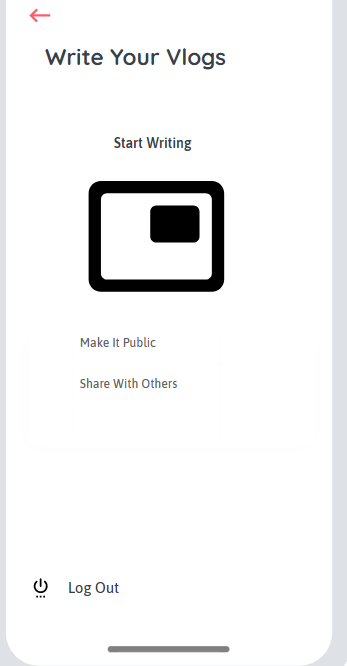
**Interface Design:**











# SYSTEM TEST PLAN

**Test Plan for Functional Requirements:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_1 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Choosing the Best Route | | | Test Execution date: 04-08-21 | | |
| Test Title: By providing destination, finding the best route from Google Map | | | | | |
| Description: Test the best route finding from Google Map | | | | | |
| Precondition (If any): User must provide the destination | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to home 2. Enter Best route 3. Provide destination 4. Click the tik mark | The best route shown from Google Map | User should find the best route | | As expected, | Pass |
| Post Condition: User finds the best route for his destination from Google Map. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_2 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Rabbi | | |
| Module Name: Finding Ideal Hotel | | | Test Execution date: 04-08-21 | | |
| Test Title: By providing destination, finding ideal hotel from Google Map | | | | | |
| Description: Test finding of ideal hotel in selected destination | | | | | |
| Precondition (If any): User need to provide the destination | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to home 2. Enter Ideal Hotel 3. Provide location 4. Click the tik mark | Ideal hotel location  shown from Google Map in selected destination | User should find the ideal hotel location | | As expected, | Pass |
| Post Condition: User finds the ideal hotel location in selected destination. | | | | | |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_3 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Nearest Transportation | | | Test Execution date: 04-08-21 | | |
| Test Title: By providing current location, nearest transportation system will be shown from Google Map | | | | | |
| Description: Test the nearest transportation finding from Google Map | | | | | |
| Precondition (If any): User must turn on GPS in the device. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to home 2. Enter Transportation 3. Provide destination 4. Click the tik mark | The nearest transportation shown from Google Map | User should find the nearest transportation | | As expected, | Pass |
| Post Condition: User finds the nearest transportation from Google Map. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_4 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Weather Update | | | Test Execution date: 04-08-21 | | |
| Test Title: By providing destination, get the weather update of that destination. | | | | | |
| Description: Test the weather finding from Global Weather | | | | | |
| Precondition (If any): User need to provide the destination | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to home 2. Enter Weather 3. Provide destination 4. Click the tik mark | Weather update shown from the Global Weather | User should get the weather update | | As expected, | Pass |
| Post Condition: User gets the weather update for the provided destination from Global Weather. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_5 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Create Event | | | Test Execution date: 04-08-21 | | |
| Test Title: User can create event where others can join. | | | | | |
| Description: Test the event creation and members interaction. | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to Create Event 2. Give a name of the event. 3. Add members 4. Request approval 5. Cancelling Request | Event created with a name. Members are interacting in the event. User controlling the members. | User should create the event successfully and managing other members. | | As expected, | Pass |
| Post Condition: User creates the event and interact with other members and manage them. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_6 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Making Groups | | | Test Execution date: 04-08-21 | | |
| Test Title: User can make group among the event members. | | | | | |
| Description: Test the group the group creation among the event members | | | | | |
| Precondition (If any): Member need to joined in the particular event | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to Make Groups 2. Add members 3. Request approval 4. Cancelling Request | Group created with event members. Members are interacting in the group. | User should create the group successfully and managing other members. | | As expected, | Pass |
| Post Condition: User creates the group and interact with other members. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_7 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Chatting | | | Test Execution date: 04-08-21 | | |
| Test Title: User can make chat with group members in personal or in group. | | | | | |
| Description: Test the user can chat with other group members in personal and in groups | | | | | |
| Precondition (If any): Member need to be present in the group | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to Chatting 2. Choose person or group 3. Make calls and do other operations of messenger. 4. Add members 5. Request approval 6. Cancelling Request | Chatting with others in personal or in group and can make all messenger operations. | User should chat with others in personal or in group. | | As expected, | Pass |
| Post Condition: User chats with other group members in personal or in group. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_8 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Writing Vlogs | | | Test Execution date: 04-08-21 | | |
| Test Title: User can write vlogs | | | | | |
| Description: Test the user can write vlogs | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to Vlogs 2. Write vlogs 3. Can make the vlog public 4. Share with others | Writing vlogs and control privacy as well as sharing with others. | User should write a vlog and control privacy also able to share it with others. | | As expected, | Pass |
| Post Condition: User writes the vlog with privacy control and sharing with others. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_9 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Gallery | | | Test Execution date: 04-08-21 | | |
| Test Title: User can make gallery and upload files | | | | | |
| Description: Test the files are uploading in the gallery | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to Gallery 2. Make folders and upload files 3. Can choose location of files 4. Add new files 5. Delete Existing files | Files uploaded in the folders of the gallery. | User should be able to upload files from choosing location and add or delete more files | | As expected, | Pass |
| Post Condition: User uploads files from choosing location and adding or deleting files from gallery. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: FR\_10 | | | Test Designed date: 21-03-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Log Out | | | Test Execution date: 04-08-21 | | |
| Test Title: User can exit the system | | | | | |
| Description: Test the log out option makes the exit way from the system | | | | | |
| Precondition (If any): User need to logged in once first. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to Log Out 2. Choose Yes or Cancel 3. If, Yes, then exit the system 4. Otherwise, stay in the system | Exit the system by choosing Yes after going to Log Out | User should exit the system by choosing Yes from Log Out | | As expected, | Pass |
| Post Condition: User exit the system. | | | | | |

**Test Plan for Non-Functional Requirements:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: NFR\_1 | | | Test Designed date: 11-07-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Availability | | | Test Execution date: 04-08-21 | | |
| Test Title: Checking the availability of the software | | | | | |
| Description: Test the availability of the software. | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Check the software is available in 24/7 | 24/7 service time of the software | User should get service 24/7 of the software | | As expected, | Pass |
| Post Condition: User experiencing 24/7 service of the software. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: NFR\_2 | | | Test Designed date: 11-07-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Performance | | | Test Execution date: 04-08-21 | | |
| Test Title: Checking the performance of the software | | | | | |
| Description: Test the performance of the software. | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Check whether every web page is downloading in 10 seconds with 60 Kbps modem connection | Every web page is downloading in 10 seconds with 60 Kbps modem connection | User should download every web page in 10 seconds with 60 Kbps modem connection | | As expected, | Pass |
| Post Condition: User downloading every web page in 10 seconds with 60Kbps modem connection | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: NFR\_3 | | | Test Designed date: 11-07-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Integrity | | | Test Execution date: 04-08-21 | | |
| Test Title: Checking the integrity level of the software | | | | | |
| Description: Test the security level of the software | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Check whether the data of users are well secure | User’s personal data is in under protection | User’s personal data should be protected and well secure | | As expected, | Pass |
| Post Condition: User’s data is protected and well secure | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: NFR\_4 | | | Test Designed date: 11-07-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Robustness | | | Test Execution date: 04-08-21 | | |
| Test Title: Checking the robustness of the software | | | | | |
| Description: Test the robustness of the software | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Check whether the software automatically exit after 10 minutes | Without interaction of 10 minutes software exit itself | User will be turn out of the system after 10 minutes of no interaction | | As expected, | Pass |
| Post Condition: The software itself exit after 10 minutes | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: NFR\_5 | | | Test Designed date: 11-07-21 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Rabbi | | |
| Module Name: Usability | | | Test Execution date: 04-08-21 | | |
| Test Title: Checking the usability of the software | | | | | |
| Description: Test the usability of the software. | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Check whether a complete request cab be submitted in 2 minutes | Creating, adding, deleting request completing in 2 minutes | User should submit a complete request in 2 minutes | | As expected, | Pass |
| Post Condition: User successfully submit a complete request in 2 minutes | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: NFR\_6 | | | Test Designed date: 11-07-21 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Rabbi | | |
| Module Name: Efficiency | | | Test Execution date: 04-08-21 | | |
| Test Title: Checking the efficiency of the software | | | | | |
| Description: Test the efficiency of the software. | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Check whether the software can give service in peak load situation | In peak load condition 30 percent of RAM space is free | User should get service smoothly in peak load condition | | As expected, | Pass |
| Post Condition: User experiencing smooth service in peak load condition | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: NFR\_7 | | | Test Designed date: 11-07-21 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Rabbi | | |
| Module Name: Reliability | | | Test Execution date: 04-08-21 | | |
| Test Title: Checking the reliability of the software | | | | | |
| Description: Test the availability of the software. | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Check whether the software fails 5 times out of 1000 test runs | The software gives 1000 test runs | The software should not fail no more than 5 times out of 1000 test runs | | As expected, | Pass |
| Post Condition: The software did not fail no more than 5 times out of 1000 test runs | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Tour Guideline System | | | Test Designed by: Rabbi | | |
| Test Case ID: NFR\_8 | | | Test Designed date: 11-07-21 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Rabbi | | |
| Module Name: Maintainability | | | Test Execution date: 04-08-21 | | |
| Test Title: Checking the maintainability of the software | | | | | |
| Description: Test the maintainability of the software. | | | | | |
| Precondition (If any): | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Check whether the user can modify information | Use modify username, password | User should be able to modify information | | As expected, | Pass |
| Post Condition: User successfully modified information | | | | | |